Docket No. AUS920010118US1

CLAIMS:

What is claimed is:

1.   A method for designing buffer and wire placement in an integrated circuit, the method comprising:

5   representing the surface of a integrated circuit design as a tile graph;

         receiving an allocation of buffer locations for selected tiles in the tile graph;

         routing nets between associated sources and sinks;

10   and

         selectively assigning buffer locations within selected tiles based upon buffer needs of the nets, wherein the nets are routed through selected tiles and assigned buffer locations using a cost minimization

15   algorithm.

2.   The method as recited in claim 1, the step of routing nets between associated sources and sinks comprises:

         constructing a Steiner tree for each net to

20   determine congested regions; and

         rerouting some of the nets in the congested regions to reduce wire congestion.

3.   The method as recited in claim 2, wherein the rerouting some of the nets comprises minimizing the cost

25   of placing a wire across a tile edge wherein the cost is the number of wires that will be crossing the tile edge divided by the number of wires still available for allocation.

Docket No. AUS920010118US1

4.   The method as recited in claim 1, further comprising:

   rerouting and reallocating some of the nets and buffers to reduce wire and buffer congestion and to
5   reduce the number of nets that have failed to meet their length constraint.

5.   The method as recited in claim 1, wherein the step of inserting buffers onto nets includes computing a cost, $q(v)$, for using a buffer in a particular tile and the
10   cost, $q(v)$, is given by the equation:

$$q(v) = \begin{cases} \dfrac{b(v)+p(v)+1}{B(v)-b(v)} & if \dfrac{b(v)}{B(v)} < 1 \\ \infty & otherwise \end{cases}.$$

6.   The method as recited in claim 5, further comprising:

   computing a cost array for buffer placement for a
15   particular net; wherein
   each possible arrangement of buffers is represented by an element in the cost array;
   each element in the array is the sum of costs, $q(v)$, for one possible arrangement of buffers; and
20   buffer placement for the particular net corresponds to cost array element having the smallest value.

7.   The method as recited in claim 6, the step of computing the cost array is performed for each net in the integrated circuit.

8.   The method as recited in claim 1, wherein the plurality of tiles is less than the total number of tiles in the tile graph.

9.   A computer program product in a computer readable
5 media for use in a data processing system for designing buffer and wire placement in an integrated circuit, the computer program product comprising:
    first instructions for representing the surface of a integrated circuit design as a tile graph;
10   second instructions for receiving an allocation of buffer locations for selected tiles in the tile graph;
    third instructions for routing nets between associated sources and sinks; and
    fourth instructions for selectively assigning buffer
15 locations within selected tiles based upon buffer needs of the nets, wherein the nets are routed through selected tiles and assigned buffer locations using a cost minimization algorithm.

10.   The computer program product as recited in claim 9,
20 the third instructions comprise:
    fifth instructions for constructing a Steiner tree for each net to determine congested regions; and
    sixth instructions for rerouting some of the nets in the congested regions to reduce wire congestion.

25 11.   The computer program product as recited in claim 10, wherein sixth instructions comprise minimizing the cost of placing a wire across a tile edge wherein the cost is the number of wires that will be crossing the tile edge divided by the number of wires still available.

Docket No. AUS920010118US1

12. The computer program product as recited in claim 9, further comprising:

  fifth instructions for rerouting and reallocating some of the nets and buffers to reduce wire and buffer

5 congestion and to reduce the number of nets that have failed to meet their length constraint.

13. The computer program product as recited in claim 9, wherein the step of inserting buffers onto nets includes computing a cost, $q(v)$, for using a buffer in a

10 particular tile and the cost, $q(v)$, is given by the equation:

$$q(v) = \left\{ \begin{array}{ll} \frac{b(v)+p(v)+1}{B(v)-b(v)} & if \frac{b(v)}{B(v)} < 1 \\ \infty & otherwise \end{array} \right\} \quad .$$

14. The computer program product as recited in claim 13, further comprising:

15 fifth instructions for computing a cost array for buffer placement for a particular net; wherein

  each possible arrangement of buffers is represented by an element in the cost array;

  each element in the array is the sum of costs, $q(v)$,

20 for one possible arrangement of buffers; and

  buffer placement for the particular net corresponds to cost array element having the smallest value.

15. The computer program product as recited in claim 14, wherein the fifth instructions are performed for each net

25 in the integrated circuit design.

Docket No. AUS920010118US1

16. The computer program product as recited in claim 9, wherein the plurality of tiles is less than the total number of tiles in the tile graph.

17. A data processing system for designing buffer and
5 wire placement in an integrated circuit, the data processing system comprising:

a representation unit which represents the surface of a integrated circuit design as a tile graph;

a buffer placement receiving unit which receives an
10 allocation of buffer locations for selected tiles in the tile graph;

an initial routing unit which routes nets between associated sources and sinks; and

a buffer association unit which selectively assigns
15 buffer locations within selected tiles based upon buffer needs of the nets, wherein the nets are routed through selected tiles and assigned buffer locations using a cost minimization algorithm.

18. The data processing system as recited in claim 17,
20 the third instructions comprise:

a Steiner tree construction unit which constructs a Steiner tree for each net to determine congested regions; and

a rerouting unit which reroutes some of the nets in
25 the congested regions to reduce wire congestion.

Docket No. AUS920010118US1

19. The data processing system as recited in claim 18, wherein rerouting unit minimizes the cost of placing a wire across a tile edge wherein the cost is the number of wires that will be crossing the tile edge divided by the

5 number of wires still available.

20. The data processing system as recited in claim 17, further comprising:

   a reallocation unit which reroutes and reallocates some of the nets and buffers to reduce wire and buffer

10 congestion and to reduce the number of nets that have failed to meet their length constraint.

21. The method as recited in claim 17, wherein the step of inserting buffers onto nets includes computing a cost, $q(v)$, for using a buffer in a particular tile and the

15 cost, $q(v)$, is given by the equation::

$$q(v) = \begin{cases} \frac{b(v)+p(v)+1}{B(v)-b(v)} & if \frac{b(v)}{B(v)} < 1 \\ \infty & otherwise \end{cases} \quad .$$

22. The data processing system as recited in claim 21, further comprising:

   a computing unit which computes a cost array for buffer

20 placement for a particular net; wherein

   each possible arrangement of buffers is represented by an element in the cost array;

   each element in the array is the sum of costs, $q(v)$, for one possible arrangement of buffers; and

25   buffer placement for the particular net corresponds to cost array element having the smallest value.

Docket No. AUS920010118US1

23. The data processing system as recited in claim 23, wherein computing unit computes the cost array for each net in the integrated circuit design.

24. The data processing system as recited in claim 17, wherein the plurality of tiles is less than the total number of tiles in the tile graph.